114CI01

國立臺北科技大學 114 學年度碩士班招生考試

系所組別：2300 資訊工程系碩士班

第一節 計算機概論 試題

注意事項：

1. 本試題共 17 題，共 100 分。
2. 不必抄題，作答時請將試題題號及答案依照順序寫在答案卷上。
3. 全部答案均須在答案卷之答案欄內作答，否則不予計分。

1. Please answer the following questions concisely and you do not need to give the reason or prove it.
   (1) Use Big-O notation to represent the growth rate of $f(n) = n^3 \log n + 2^n\sqrt{n}$. (2 pts)
   (2) Give the asymptotic bound on the recursive function $T(n) = 7T\left(\frac{n}{5}\right) + 8n^5$, where $T(n) = 1$ when $n = 1$. (2 pts)
   (3) What is the rule (or principle) for managing the data items in a stack? (2 pts)
   (4) What is the maximum possible number of edges in a directed graph with n nodes (vertices)? (2 pts)
   (5) What are the possible degrees of internal nodes (except the root) in a B-tree of order 9? (2 pts)

2. Mark by **True** or **False** each of the followings:
   (1) With Polish (or prefix) notation, the parentheses are necessary to override the operator precedence. (2 pts)
   (2) Removing at the tail of a singly linked list needs O(1) time in worse case. (2 pts)
   (3) If T is a proper binary tree and let e and i denote the numbers of external and internal nodes, respectively, then $e=i+1$. (2 pts)
   (4) There is an efficient way to construct a binary heap with a collection of n entries in linear time. (2 pts)
   (5) Suppose problem A can be reduced to problem B in linear time. If problem B can be solved in $O(n^2)$, then problem A can be solved in $O(n^2)$. (2 pts)

3. Consider the entries with key values 42, 21, 38, 62, 51, 18, and 8.
   (1) Suppose we would like to use the hash function $f(i) = (3i + 2) \bmod 7$ to build a 7-entry hash table as an index over the entries on the keys in the same order given above. Please draw the resulting hash table that uses *linear probing* strategy to handle collisions. (4 pts)
   (2) Instead of the hash table, we now would like to construct an AVL tree for the entries with keys in the same order given above. Please show the resulting AVL tree. (3 pts)
   (3) On the constructed AVL tree in (2), which traversal can be used to list all the keys in an increasing order? Please also provide the reason. (3 pts)

4. Please answer the following questions about the graph G in Figure 1.

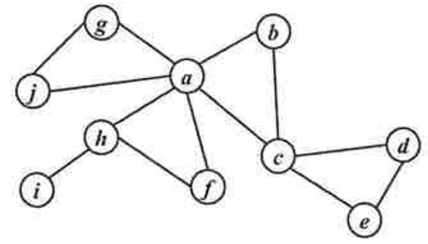


Figure 1. The graph G for problem 4

   (1) How many *biconnected components* are in G? (3 pts)
   (2) Please give the *articulation points*. If there is no articulation point, please use ∅ to denote it. (3 pts)
   (3) Suppose we are performing a BFS traversal starting with vertex a in alphabetical order. Please show the corresponding BFS-tree. (4 pts)

5. For the following questions regarding *process and thread management*, please indicate whether each statement is true or false. If a statement is incorrect, please explain the reasons. (not just correcting the errors)
   (1) Shortest job first (SJF) scheduling algorithm will not result in starvation. (2 pts)
   (2) A multithreaded program using multiple *user-level* threads can achieve better performance on a multiprocessor system than on a single-processor system. (2 pts)
   (3) The major performance gain from adding additional CPU cores to a system is mostly determined by the number of cores. (2 pts)

6. Answer the following questions regarding *synchronization*.
   (1) What's the problem with *spinlocks*? (1 pt)
   (2) What is the idea of *semaphore*? Please compare its differences from mutex locks. (3 pts)

注意：背面尚有試題

7. Among the following statements about *memory management*, please indicate whether each statement is true or false. If a statement is incorrect, please explain the reasons. (not just correcting the errors)
   (1) Pure segmentation has the problem of internal fragmentation, while pure paging has the problem of external fragmentation. (2 pts)
   (2) With demand paging, it's possible to run programs that are larger than the physical memory. (2 pts)

8. Answer the following questions regarding two disk block allocation methods for files (including *linked allocation* and *indexed allocation*) in the implementation of *file systems*.
   (1) What are the ideas of these two methods? (1 pt)
   (2) Please compare their differences, and give at least one example in real file systems. (2 pts)
   (3) Please compare their advantages and disadvantages. (3 pts)

9. An organization has been assigned the IP address block 192.168.1.0/24. The network needs to be divided into 4 subnets. Calculate:
   (1) The subnet mask for each subnet. (3 pts)
   (2) The range of IP addresses for each subnet. (3 pts)

10. A file of size 5 GB is transferred over a network link with a bandwidth of 100 Mbps and this network link has a propagation delay of 10 ms and a transmission rate of 50 Mbps, calculate the time taken to transmit this 5GB file. (3 pts)

11. Which of the following statements about Zero Trust are correct? (Multiple selection question, 2 pts)
    (A) Zero Trust means no need to verify all the time.
    (B) Provide only necessary permissions.
    (C) No need to maintain network visibility.
    (D) Zero Trust Design based on the principle of "never trust, always verify".

12. Which of the following are key principles of Zero Trust architecture? (Multiple selection question, 2 pts)
    (A) Continuous authentication and verification
    (B) Trusting devices within the corporate network
    (C) Micro-segmentation to limit lateral movement
    (D) Allowing full access once logged in

13. Given the routing table below, calculate the next hop for a packet destined for 192.168.4.25: (3 pts)

Destination	Subnet Mask	Next Hop
192.168.4.0	255.255.255.0	Router A
192.168.0.0	255.255.0.0	Router B
Default	-	Router C

14. Given the following C code:

```
#include <stdio.h>
#include <stdlib.h>
#define SIZE 50
void myStrcpy(char *dest, const char *src) {
    for (;*src != '\0'; dest++, src++)
        *dest = *src;
    *dest = *src;
}
void copyInput(char *input) {
    char buffer[SIZE];
    myStrcpy(buffer, input);
    printf("Buffer contains: %s\n", buffer);
}
int main() {
    char *Input = (char*) malloc(sizeof(char)*SIZE);
    gets(Input);
    copyInput(Input);
    return 0;
}
```

(A) Identify **where** the vulnerability occurs and explain **how** it can be exploited. (2 pts)
(B) Propose **modify strategies** for each issue. (2 pts)

15. The following table shows the number of **instructions** for a program. Assuming that the **CPI** for the **R-format, Load, Store, Branch, and Jump** instructions is **1, 5, 4, 3** and **2, respectively.**

R-Format	Load	Store	Branch	Jump
50×10^9	20×10^9	10×10^9	40×10^9	30×10^9

   (1) Find the average CPI for the program. (3 pts)
   (2) By how much we can improve the CPI of Load instructions if we want the program to run 1.25 times faster? (3 pts)

16. Translate the following C to MIPS code.

```
int swap(int a[], int m, int n, int size) {
    int i, j, sum;
    sum = 0;
    for (i = m, j=n; i<size; i++, j++) {
        sum = sum + a[i+1] - a[j+2];
    }
    return sum;
}
```

```
swap:
    addi $sp, $sp, -12#   sum, i, j, temp
    sw    $ra, 12($sp)  # store ra
    sw    $s0, 8($sp)  #    sum
    sw    $t0, 4($sp)  #    i
    sw    $t1, 0($sp)  #    j
    add   $s0, $zero, $zero# sum = 0
    add   $t0, $a1, $zero#    i = m
    add   $t1, $a2, $zero#    j = n

loop:
    slt   $t3, $t0, $a3#    i < size
    beq   $t3, $zero, end_swap#    i >= size, end
    #   (1) sum = sum + a[i+1] - a[j+2];
    sll $t2, $t0, 2                # i * 4
    sll $t4, $t1, 2                # j * 4
    _(I)_ $t2, $t2, _(III)_        # a[i] address
    _(II)_ $t4, $t4, _(IV)_        # a[j] address
    _(V)_ $t5, 4(_(VI)_)           # a[i+1]
    (VII) $t6, 8((VIII))           # a[j+2]
    add $s0, (IX) , (X)            # sum = sum + a[i+1]
    sub $s0, $s0, $t6              # sum = sum - a[j+2]
    addi $t0, $t0, 1               # i++
    addi $t1, $t1, 1               # j++
    j     loop

end_swap:
    add   $v0, $s0, $zero          # return sum
    lw    $ra, 12($sp)             # restore ra
    lw    $s0, 8($sp)              # restore sum
    lw    $t0, 4($sp)              # restore i
    lw    $t1, 0($sp)              # restore j
    addi $sp, $sp, 12              # release space
    jr    $ra
```

Multiple selection question:

(1) (A) (I) is add; (B) (II) is lw; (C) (III) is $a1; (D) (IV) is $a0; (E) (X) is $t6. (3 pts)

(2) (A) (V) is lw; (B) (VI) is $t2; (C) (VII) is add; (D) (VIII) is $t5; (E) (IV) is $t0.(3 pts)

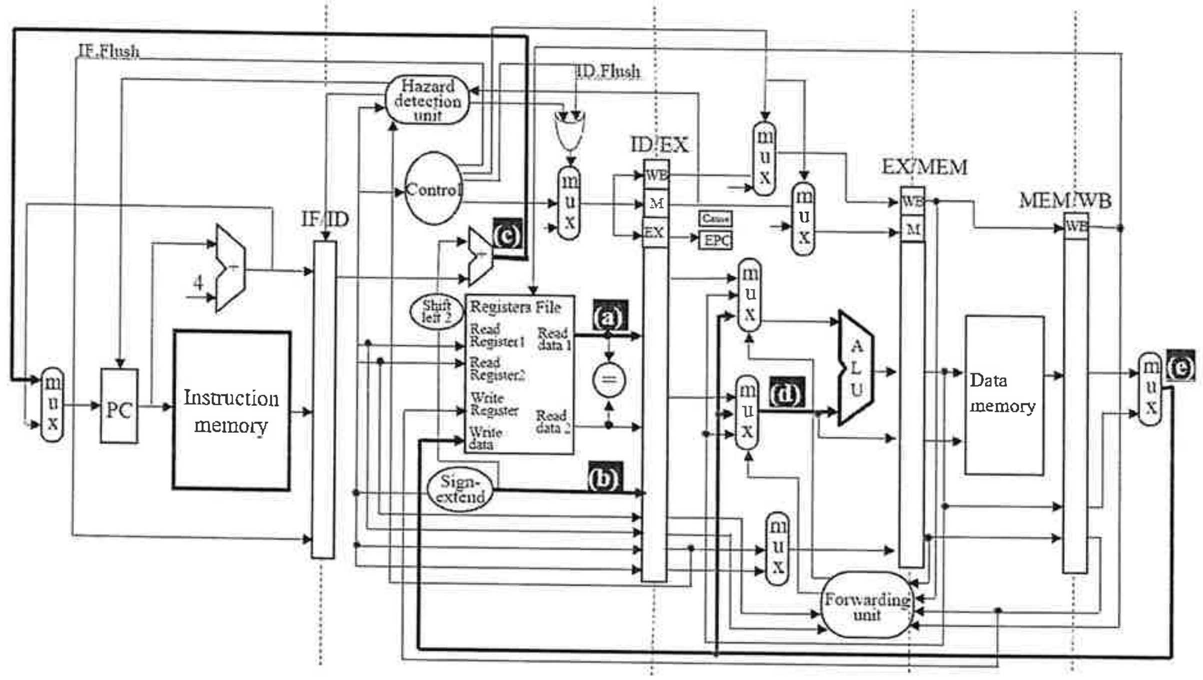
(3) (A) (II) is add; (B) (IX) is $s0; (C) (VII) is lw; (D) (II) is sw; (E) (V) is add. (2 pts)

17. Assume the address of the first instruction is 3000, and that all values in Memory are 0, and values of register are the following Table.

r0	r1	r2	r3	r4	r5
4	40	0	2	3	1

PC	instruction
3000	loop: lw r5, 4(r1)
3004	add r2, r2, r3
3008	sub r3, r4, r5
3012	beq r4, r0, loop

Consider the above instructions executing in the processor implemented by the following Figure. Please fill in the decimal value of (a), (b), (c), (d), (e) in the first five cycles. If instruction in the current stage is not used, fill in X. (Hint: For the output of each MUX, you should determine it through the instruction in the current Stage.)



	(a)	(b)	(c)	(d)	(e)
Cycle 1	X	X	X	X	X
Cycle 2	40	4	X	X	X
Cycle 3	0	X	X	4	X
Cycle 4	(I)	(II)	X	(III)	X
Cycle 5	(IV)	(V)	(VI)	(VII)	(VIII)

Multiple Choice:

(1) (A) (I) is 0; (B) (II) is 0; (C) (III) is 2; (D) (IV) is 1; (E) (V) is 0. (2 pts)

(2) (A) (V) is 4; (B) (VI) is X; (C) (VII) is 0; (D) (VIII) is 1; (E) (I) is X. (2 pts)

(3) (A) (II) is 1; (B) (VI) is 3000; (C) (VII) is X; (D) (VIII) is 0; (E) (IV) is 40. (2 pts)